IN THE SPECIFICATION:

Please replace paragraph **[0003]** with the following amended paragraph:

[0003] Error correcting codes are well known. Early codes included parity codes and block codes where syndromes are generated to help determine whether the received data stream at a receiving device is the same as the data stream that was sent by a transmitting device. Parity codes operate at the byte or word level while block codes operated on relatively large (typically, up to a couple of hundred bytes) message blocks. Recently, convolutional codes have enjoyed increasing popularity in both magnetic recording applications and communication systems, such as optical, wireline and wireless communication systems. In contrast to block codes, convolutional codes operate on serial data, one or a few bits at a time. The Viterbi algorithm is a widely used convolutional code <u>decoding</u> algorithm and is of the type known as a maximum-likelihood decoding algorithm.

Please replace paragraph **[0005]** with the following amended paragraph:

[0005] The convolutional encoder employs an algorithmic method to generate an encoded output stream that is based on the current input and a selected number of previous inputs to the encoder. The convolutional encoder may be described [[as]] <u>using</u> a code tree with each digital input specifying a branch on the tree. In the tree view, each symbol (zero or one) drives the encoder to a different state.

Please replace paragraph **[0006]** with the following amended paragraph:

[0006] After transmission of the encoded data stream to the receiving device, a decoder process operates on the encoded data stream to recover the original data stream from the encoded data stream. The decoder determines the original inputs to the encoder by ~~taking advantage of the physical properties~~ <u>exploiting the trellis structure</u> of the code ~~generator~~ to determine the maximum likelihood path through the tree ~~to~~

~~converge on the actual input~~ given the received sequence. More specifically, based upon the encoded input data, the constraint length (k) and the generator polynomials, a cost or path metric is computed for each possible branch of the tree diagram for the given decoder sample. The path-metric is the cost of traversing the tree to the current point. The computed branch-metric is then added to an accumulated path metric. The [[low]] lower cost updated path for a given state is selected and stored as a selection bit. This selection bit is used by path-metric comparison logic to specify which of the two possible paths is more likely.

Please replace paragraph **[0007]** with the following amended paragraph:

[0007] For each input sample, the cost at each node of the tree is computed and the ~~low~~ lower cost path is chosen as a survivor. The selection bit for each of the nodes is stored in the survivor memory at a specific location based upon which node was being computed. By storing the selection bits in the survivor memory, a history of the most likely transitions into every given state for each input sample is retained. Once enough data is stored in the survivor memory, the output stage is activated. Although there are a number of different methods to recover the data stream from the survivor data. One of the most common techniques used for the ~~output device~~ recovery is the trace-back method. In the trace-back method, the stored survivor data corresponds to the tree connections computed by the decoder. Starting from the latest data point, the path through the code tree is reconstructed in reverse order so that the last bit out of trace-back corresponds to the very first data bit input to the encoder. Due to inherent properties of the convolutional code, the survivor data eventually merges to the maximum likelihood sequence of inputs. The number of samples required for this merging is called the convergence interval of the decoder. The traced back sequence after the convergence interval is the time reversed input sequence.

Please replace paragraph **[0013]** with the following amended paragraph:

[0013] In accordance with an embodiment of the present invention, an error correcting system and method for decoding convolutional codes is provided. More specifically, an efficient and novel method is disclosed for decoding convolutionally encoded data using a maximum-likelihood decoding algorithm. In one preferred embodiment, received encoded data is decoded by a Viterbi decoder that incorporates a pipelined architecture to improve the throughput rate of [[a]] the Viterbi decoder. Advantageously, the decoder is not susceptible to stalls, delays, dropouts or interruption in delivering the decoded received data stream to the user. The present invention provides a simple yet elegant solution to implementing a pipelined trace-back solution for a Viterbi decoder that not only eliminates possibility of stalls or delays but that also overcomes hardware limitations inherent in a pipeline architecture while providing a high performance communication system.

Please add paragraph **[0028.1]** with the following paragraph:

[0028.1] Figure 11 illustrates convolutionally encoded data in accordance with an embodiment of the present invention.

Please replace paragraph **[0038]** with the following amended paragraph:

[0038] FIG. 3 illustrates a tree representing the allowable state transitions for an input data stream ~~because~~ wherein convolutionally encoded data is decoded through knowledge of the possible state transitions created from the dependence of the current symbol on past data. To illustrate the progression through the tree, let the encoder state at time t be denoted by state(t), which is not to be confused with the actual state value. Assume that at an initial state (i.e., state (0)), the shift register has been previously initialized with all zeros at each register location and a first input bit is shifted into location Tb0 (FIG. 2). This input bit may be either a `0` or a `1`. In the case where the input bit is a `0`, then for the next state, state (1), the shift register will still have the value of `0`. If, however, the input bit is a `1`, then for state (1), the value of the shift

register will be 32. When the next input bit is shifted into the shift register, the first bit will be shifted to Tb1 and a new bit shifted into register Tb0.

Please replace paragraph **[0040]** with the following amended paragraph:

[0040] A Viterbi decoder [[is]] uses an algorithm that obtains a maximum likelihood sequence estimate (MLSE) from a convolutionally encoded received data stream. The Viterbi decoder typically is based upon certain parameters referred to as the trace-back length (TL), the decoding length (DL) and the convergence length (CL) where the relationship between these parameters is:

TL=DL+CL (1)

Please replace paragraph **[0042]** with the following amended paragraph:

[0042] Figure 4 is a block diagram of a Viterbi convolutional code~~, or Viterbi,~~ decoder 112. Decoder 112 includes a branch metric calculation module 402 that accepts encoded input data streams. Module 402 computes a cost, based upon the encoded input data, the constraint length (k) and the generator polynomials, for each possible branch of the tree diagram for the encoded input data stream. The branch metrics ~~represents~~ represent the cost of traversing along a specific branch. As the calculations are completed, the computed cost is passed to a path metric calculation module 404.

Please replace paragraph **[0056]** with the following amended paragraph:

[0056] Rather than accept the conventional wisdom that the pipeline will result in performance degradation[[,]] of the trace-back phase, the present invention assumes that with two pipeline stages, the present invention selects a cutout or portion of the code trellis. Memory is then organized into several trellis cutouts that provide a memory trellises and organizing memory based upon the sub-trellises may require the memory to expand in order to accommodate duplication of states.

Please replace paragraph **[0058]** with the following amended paragraph:

[0058] In operation, a 32-bit memory word is loaded into the pipeline register. This word is then transferred to the mux and one bit is selected as representing the most likely path. However, because of the inserted pipeline, the data out of the memory is no longer in sync with the selection process. The first word generates an address back into the memory. One cycle after that, the word designated by the address must be in the mux~~. And~~ <u>and</u> that word is then going to be shifted and used as the selector of the next word to come out of memory. However, there is now a loss of synchronization due to the addition of the pipeline. Compounding the problem is the fact that memory must be accessed twice resulting in the output data rate being reduced by a factor of two. Rather than accept the fact there will be stalls due to the lack of knowledge of the trace-back path and system parameters, the pipeline and the memory re-organization eliminates stalling so that the decoder does not need to shut down until it can perform the necessary computations.

Please add paragraph **[0060.1]** with the following paragraph:

[0060.1] In summary, what is disclosed herein and specifically described in the code beginning at paragraph [0062] is a method for implementing a decoder based on a pipelined architecture which includes the steps of receiving convolutionally encoded data 1110 and generating a tree for that encoded data. The convolutionally encoded data is then decoded for each received bits by determining a trace-back length at 1112 obtaining a trellis diagram for the convolutional encoded that generated the encoded received data stream 1114. For each bit and set that received data stream, the plurality of forward butterfly computations determines survivor path bits 1116. For each butterfly computation the resulting survivor path bits for each state are stored in a trace-back memory 1118. The above steps 1110 – 1118 are repeated until all bits in the encoded received data stream have been recorded 1120. When the bits have been decoded and recorded, a trace-back window is selected 1122. The method then sequentially

decumbence by two 32 bit word steps to access the trace-back memory 1124. A trace bit is used to perform a look at function to determine a computed address of a future survivor word 1126. The future survivor word is then determined to be in an odd or even state 1128. A decoded bit is then determined from a computed address 1130. The decoded data stream corresponding to the encoded received data stream is then output 1132.

Please replace paragraph **[0071]** with the following amended paragraph:

[0071] Additionally, any signal arrows in the drawings/Figures should be considered only as exemplary, and not limiting, unless otherwise specifically noted. Furthermore, the term "or" as used herein is generally intended to mean "and/or" unless otherwise indicated. Combinations of components or steps will also be considered as being included ~~noted~~, where terminology is foreseen as rendering the ability to separate or combine ~~is unclear~~.